

VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

KATEDRA INFORMATIKY

Systém pro záznam IP Televize

System for Recording IP Television

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 7. května 2010

Tomáš Komenda

Rád bych poděkoval vedoucímu této bakalářské práce Ing. Davidu Seidlovi za jeho připomínky, cenné rady a čas, který mi věnoval při řešení problémů během psaní této práce.

ABSTRAKT

Tato práce pojednává o návrhu a implementaci systému, který umožňuje nahrávat a ukládat televizní pořady vysílané prostřednictvím digitálního a internetového televizního vysílání. Účelem programu je především umožnit nahrávání z více různých zdrojů najednou a několika uživatelům zároveň, ale také usnadnit a zpříjemnit uživatelům proces přípravy nahrávání prostřednictvím funkcí, jako je například zobrazování informací o vybraných pořadech nebo aktivní vyhledávání dalších dílů seriálů. Systém byl při programování rozdělen do dvou částí; v dokumentu je rozebrána instalace a konfigurace každé z těchto částí i jejich vzájemná spolupráce.

KLÍČOVÁ SLOVA: digitální vysílání, systém DVB, systém IPTV, internetová televize, streaming, multicasting, OS Linux, programovací jazyk PHP, uživatelský systém, nahrávání videa, ukládání videa.

ABSTRACT

This thesis discusses the design and implementation of the system that allows you to record and store TV programs that are transmitted by digital or internet television. The main purpose of the system is not only to allow recording from multiple sources at once and multiple users simultaneously, but also to make easier and more comfortable the process of preparing recording to users using the functions, such as displaying the information about selected programs or automatic searching for the next episode of the program. The system was divided into two parts while programming; in this document there is described the installation and configuration each of these parts and their cross cooperation.

KEY WORDS: digital broadcasting, DVB system, IPTV system, Internet television, streaming, multicasting, OS Linux, PHP programming language, user system, video recording, video storage.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

DVB	Digital Video Broadcasting
DVB-C	Digital Video Broadcasting Cable
DVB-S	Digital Video Broadcasting Satellite
DVB-T	Digital Video Broadcasting Terrestrial
EPG	Electronic Program Guide
FEC	Forward Error Correction
HTML	HyperText Markup Language
IGMP	Internet Group Management Protocol
IPTV	Internet Protocol Television
MMS	Microsoft Media Server
OS	operační systém
PAT	Program Association Table
PHP	programovací jazyk
PID	identifikátor paketu
PIM	Protocol Independent Multicast
PMT	Program Map Table
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase-Shift Keying
RTP	Real-time Transport Protocol
RTSP	Real Time Streaming Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
XML	Extensible Markup Language

OBSAH

1 Úvod.....	3
2 Teoretická východiska	4
2.1 Systém DVB	4
2.1.1 Systém DVB-T.....	4
2.1.2 Systém DVB-S.....	7
2.1.3 Systém DVB-C.....	7
2.2 Systém IPTV.....	7
2.3 Internetová televize	8
2.4 Streaming.....	9
2.4.1 Unicasting	10
2.4.2 Broadcasting.....	11
2.4.3 Multicasting	11
3 Požadavky na systém	13
3.1 Funkce systému.....	13
3.2 Prostředí systému	13
4 Návrh implementace	14
4.1 Část uživatelského systému	14
4.1.1 Programovací jazyk PHP	14
4.1.1.1 Některé vlastnosti jazyka PHP	15
4.1.1.2 Výhody PHP.....	15
4.1.1.3 Nevýhody PHP	16
4.1.2 Určení aktérů.....	16
4.1.3 Seznam a popis procesů.....	16
4.1.4 Plánování nahrávek	18
4.1.5 Struktura datových souborů	19
4.1.6 Vzhled a vnitřní řešení uživatelského systému	21

4.2 Část ukládání videa	21
4.2.1 Rozbor programu.....	21
4.2.2 Průběh programu	22
4.3 Spolupráce Části uživatelského systému s Částí ukládání videa.....	22
5 Instalace projektu.....	24
5.1 Instalace a konfigurace Části uživatelského systému	24
5.2 Instalace a konfigurace Části ukládání videa	25
6 Závěr	26
Seznam použité literatury.....	27
Seznam obrázků.....	28

1 ÚVOD

Televizní stanice na celém světě vysílají denně stovky zajímavých pořadů. Jelikož není v lidských silách shlédnout je všechny v reálném čase, je jediným řešením jejich nahrávání.

S rozvojem digitálního a internetového televizního vysílání se čím dál více diváků přesunuje od televizních obrazovek k počítačovým monitorům, proto je potřeba nahradit klasické videorekordéry softwarem, který je dokáže zastoupit a v mnohém i překonat. Takový program by například mohl umět nahrávat více pořadů najednou a z různých zdrojů a to i nezávisle pro více uživatelů. Měl by umět zobrazovat informace o vybraných pořadech a aktivně vyhledávat další díly oblíbených seriálů. Tímto i dalšími funkcemi, jako je například nastavení nahrávání jediným kliknutím myši, by měl svým uživatelům maximálně usnadnit a zpříjemnit a na minimum zkrátit nezáživné nastavování nahrávání.

Hlavním cílem mé bakalářské práce je pokusit se takový software vytvořit. Dalším cílem je, aby tento program měl co nejširší možnosti využití a aby byl maximálně flexibilní. Měl by mít co nejmenší nároky na použité hardwarové i softwarové vybavení.

2 TEORETICKÁ VÝCHODISKA

2.1 Systém DVB

DVB (Digital Video Broadcasting) je sbírka mezinárodních standardů pro digitální televizní vysílání. Standardy DVB jsou udržovány mezinárodním průmyslovým společenstvím DVB Project, které má více než 270 členů z různých oborů a je držitelem mnoha patentů v oblasti digitalizace a přenosu obrazových i zvukových záznamů, včetně například kódování videa MPEG. Systém DVB je tedy technologie pro přenos video či audio signálu v digitální podobě. Zdigitalizovaný obraz a zvuk je přenášen v datovém kanálu, multiplexu, který může přenášet různý počet televizních programů v závislosti na jejich kvalitě. Mimo obraz a zvuk mohou být přenášena ještě metadata různých datových služeb, jako je například teletext, titulky nebo informace o televizním programu EPG (Electronic Program Guide). Existuje i nepříliš rozšířená odnož DVB-RC, která obsahuje i zpětné kanály, čímž dává divákům možnost interakce.

Systém DVB využívá k přenášení dat různá média, což předurčuje základní rozdělení jeho odnoží podle toho, zda je signál šířen kabelem nebo pozemním či satelitním vysíláním.

Tyto základní distribuční systémy vycházejí ze stejného základu. Všechny používají pro přenos dat transportní proud MPEG. To je komunikační protokol pro přenos audia, videa i dat. Jedná se o formát digitálního kontejneru, který zapouzdřuje několik elementárních paketových proudů, typicky proud audia a proud videa. Mimo tyto proudy může přenášet ještě i jiná data, čehož se využívá pro přenos různých datových služeb. U většiny systémů DVB se využívá transportní proud MPEG-2, což je základní standard, který využívá ztrátovou kompresi pro kódování videa a příslušného audia nejen u systémů DVB, ale například také u distribuce filmů na discích DVD.

Rozdíly mezi základními distribučními systémy DVB vycházejí zejména z technických omezení daného přenosového média. Systémy proto používají různá modulační schémata a opravné kódy.

2.1.1 Systém DVB-T

DVB-T (Digital Video Broadcasting – Terrestrial) je pozemní digitální vysílání, které v ČR postupně nahrazuje v současné době končící analogové televizní vysílání. V DVB-T je obrazový i zvukový proud televizního programu digitalizován a spolu s dalšími datovými službami je s využitím transportního proudu MPEG-2 přenášen v multiplexu. V jednom multiplexu může být přenášeno několik televizních programů, přičemž jednotlivé programy nemusejí mít přiřazenu stejnou šířku přenosové kapacity. Díky technologii Statistický multiplex totiž dochází k dynamickému přizpůsobování datového toku programu a tím k přizpůsobování jeho kvality. To umožňuje operativní přidělení vyššího datového toku programu, který jej momentálně potřebuje, na úkor programu, který jej plně nevyužívá. V případě, že jeden pořad potřebuje větší rozlišení, rozšíří se jeho datový tok, zatímco když zrovna větší rozlišení nepotřebuje, datový tok se opět zúží. Vzhledem k tomu, že proměnný datový tok se nastavuje

pro každý program zvlášť, může v případě, kdy všechny programy multiplexu v jednom okamžiku potřebují větší rozlišení, dojít k degradaci kvality obrazu.

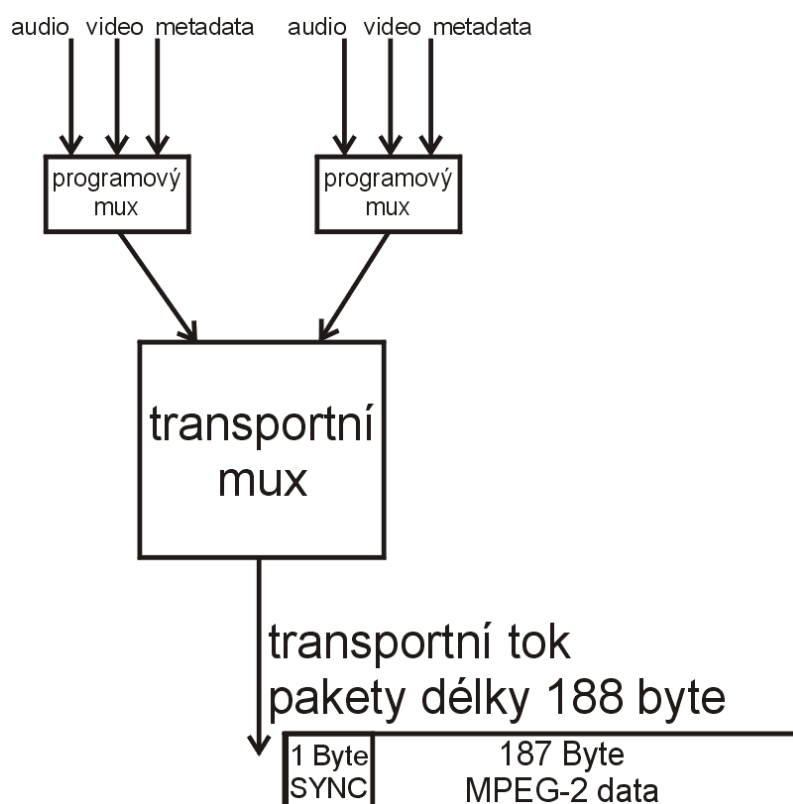
U analogového pozemního vysílání je z blízkých vysílačů vysílán určitý televizní program vždy na různých kanálech. To proto, aby nedocházelo k překrytí signálů z různých vysílačů na jednom přijímači. V takovém případě by totiž signál ze vzdálenějšího vysílače dorazil k přijímači se zpožděním, což by způsobilo na obrazovce přijímače poruchy obrazu známé jako „duchové“. Naproti tomu pozemní vysílání DVB-T podporuje synchronní provoz, což znamená, že každý multiplex může být vysílán ze všech vysílačů na stejném kanále. To představuje velkou úsporu kmitočtového pásma. Synchronní provoz je umožněn díky COFDM modulaci, která používá ochranný časový interval. Ten zajišťuje, aby nikdy nedošlo k situaci, kdy k přijímači mezi dvěma vysílači dorazí nějaký datový paket z bližšího vysílače dříve, než předchodí datový paket z nějakého vzdáleného vysílače. Délka časového ochranného intervalu je volitelná a závisí zejména na dané geografické situaci.

Zatímco u analogového vysílání způsobuje příjem jednoho programu z více vysílačů vždy problém, u vysílání digitálního je tomu právě naopak. Pokud totiž digitální přijímač přijímá nekvalitní signál z několika vysílačů, získá vinou nekvalitního signálu několik verzí poškozených dat, která dokáže vzájemně zkombinovat, čímž může získat kvalitní obraz.

Paket (viz obrázek č. 1) je základní datovou jednotkou transportního proudu. Jeho délka je 188 bytů. K tomu se většinou přidává ještě určitý druh samoopravného kódu v závislosti na médiu. V případě DVB-T je proto skutečná délka paketu 204 bytů. Prvních 32 bitů paketu je klíčových, tvoří hlavičku (viz obrázek č. 2), která je velmi důležitá. Hlavička sestává ze synchronizačního bytu 0x47, následovaného třemi příznakovými bity. Za nimi následuje další velmi podstatná část – identifikátor paketu PID, který udává, ke kterému proudu daný paket patří. Pro každý program multiplexu je vysílána tabulka PMT (Program Map Table), která určuje, jaké hodnoty PID k danému programu patří; typicky jedna hodnota PID pro proud video, jedna až dvě pro proud audio, případně další pro data dalších služeb. V tabulce PAT (Program Association Table) je potom seznam všech programů, které jsou v daném multiplexu k dispozici, kde každý program je identifikován 16bitovou hodnotou program_number. Každý program v tabulce PAT je asociován hodnotou PID k tabulce programu PMT.

Při dekódování určitého televizního programu je tedy program nalezen v tabulce programů PAT, zde je zjištěn PID daného programu, podle kterého se najde správná tabulka PMT, kde se zjistí PID čísla proudu video a audio daného programu, se kterými se porovná PID každého přijatého paketu a pouze pokud souhlasí, je paket zpracován.

Obrázek č. 1: Paket transportního proudu



Obrázek č. 2: Hlavička paketu transportního proudu

Název	Počet bitů
sync byte	8
Transport Error Indicator (TEI)	1
Payload Unit Start Indicator	1
Transport Priority	1
PID	13
Scrambling control	2
Adaptation field exist	1
Payload data exist	1
Continuity counter	4
Adaptation field	0 a víc
Payload data	0 a více

2.1.2 Systém DVB-S

DVB-S (Digital Video Broadcasting Satellite) je standardem pro digitální vysílání a příjem televizního signálu prostřednictvím satelitu. Systém DVB-S musí zohledňovat zvláštnosti satelitního přenosu, kterými je zejména vysílání na velkou vzdálenost a z toho plynoucí nízká úroveň signálu na straně přijímače. Proto se využívá kombinace několika postupů, které mají zaručit kvalitu signálu zejména díky samoopravnému kódu standardu FEC (Forward Error Correction), který pracuje na principu přidávání redundantních dat, která jsou využita v případě výskytu chyby přenosu. Signál je modulován čtyřfázovým digitálním modulačním schématem QPSK (Quadrature Phase-Shift Keying).

2.1.3 Systém DVB-C

DVB-C (Digital Video Broadcasting Cable) je standardem pro vysílání a příjem digitální televize po kabelu. Systém je kódován s využitím transportního proudu MPEG-2 a k modulaci signálu využívá systém QAM (Quadrature Amplitude Modulation), nejčastěji ve verzi 64QAM, která nabízí přenosovou rychlost 38,47Mbit/s při šířce pásma 8MHz. Taková přenosová rychlost postačí pro přenos asi deseti televizních programů.

2.2 Systém IPTV

IPTV je zkratka pro Internet Protocol Television neboli Televizi přes internetový protokol. Jedná se o systém, který přenáší služby digitální televize prostřednictvím IP protokolu a počítačové sítě. Tato služba bývá velmi často součástí dodávky širokopásmového připojení. Pro domácí uživatele je IPTV také často poskytována jako součást služby Video on Demand (VoD), která nabízí uživatelům možnost výběru určitého pořadu, filmu či seriálu z dané nabídky poskytovatele a následné okamžité shlédnutí. Narozdíl od internetové televize, kterou mohou zpravidla sledovat všichni účastníci internetu, IPTV bývá dodávána většinou v uzavřené síťové infrastruktuře nebo v lokálních sítích různých poskytovatelů. Nejde tedy o klasické internetové vysílání. IPTV putuje po privátní IP síti, která není dostupná široké veřejnosti, ale pouze těm zákazníkům, kteří si ji předplatili. Pro používání IPTV musí být uživatel vybaven buď osobním počítačem, nebo set-top boxem připojeným k TV. K přenosu videa se používá transportní proud MPEG-2, případně MPEG-4, a technologie IP Unicast pro přenos videa k jedinému uživateli (v případě VoD) nebo IP Multicast pro přenos videa k mnoha uživatelům zároveň (v případě živého vysílání).

Celá architektura IPTV vypadá tak, že většinou existuje nějaký centrální server, který získává potřebné video kanály. Ty může získávat z různých zdrojů, jako je digitální nebo i analogové pozemní vysílání, satelit nebo různé vlastní knihovny či archivy pořadů. Z těchto zdrojů se vytvoří datové proudy – streamy. Pro každý televizní kanál se vytvoří jeden audiovizuální datový proud. Všechny tyto datové proudy jsou následně přenášeny po páteřní síti provozovatele co nejblíže k uživatelům. V místě, kde sdílená páteřní síť přechází do přístupové sítě, dochází k rozvětvení datových proudů. Od tohoto místa se již přenášejí datové proudy pouze k individuálním uživatelům, kteří se na ně v daném okamžiku chtějí dívat. Vzhledem

k technickým omezením přenosového kanálu je většinou možné přenášet pro jednoho uživatele pouze jeden televizní kanál. V případě, že se více uživatelů dané podsítě chce ve stejném okamžiku dívat na stejný kanál, by nebylo hospodárné přenášet vedle sebe několik stejných streamů. Proto se pro rozvětňování datových proudů ke koncovým uživatelům využívá multicast. Každý jednotlivý televizní kanál potom představuje jednu multicastovou skupinu a konkrétní uživatelé, kteří chtějí právě sledovat daný kanál, jsou členy příslušné multicastové skupiny. V okamžiku, kdy se uživatel rozhodne přepnout kanál, je ze stávající multicastové skupiny odhlášen a přihlášen do jiné skupiny, která náleží přepnutému kanálu. Tato technologie přepínání programů se nazývá Channel zapping a je zdrojem určitého nepohodlí při používání IPTV. Přejed uživatel z jedné multicastové skupiny do druhé je totiž operace, jejíž realizace určitou dobu trvá, což způsobuje určité zpoždění při přepínání programů. Průměrná doba přepnutí programu může dosahovat až několika sekund, což už není zanedbatelné.

2.3 Internetová televize

Internetová televize (známá také pod názvy Internet TV, iTV nebo Online TV) je služba nabízející sledování videa prostřednictvím internetu. Uživatelům nabízí možnost sledovat jak živé vysílání televizních kanálů, tak i vybrané pořady z nabídky provozovatele služby. Tato služba je často využívána zpravodajskými kanály k živému šíření jejich vysílání. Komerční televize zase často využívají této služby, aby svým divákům nabídly možnost sledování svých pořadů či seriálů z archivu. Hudební televize mohou nabídnout možnost shlédnutí všech videoklipů svých žebříčků, a tak dále. Existují i komerční a hlavně amatérské internetové televize, které nabízejí svým divákům vlastní program výhradně prostřednictvím internetu. Při malé sledovanosti a nevelkých nárocích na kvalitu se totiž taková internetová televize dá provozovat s velmi malými náklady.

Přístup k internetové televizi je velmi jednoduchý. Divákovi stačí jen mít dostatečně rychlé připojení k internetu a do internetového prohlížeče zadat webovou adresu televize, kterou si přeje sledovat. Na zobrazené stránce se divákovi kromě informací ve formě psaného textu zobrazí také okénko přehrávače, ve kterém může přímo sledovat vysílání. V některých případech je, v závislosti na technologii, kterou daný server používá, nutné doinstalovat do prohlížeče podporu (plugin) přehrávače, který dekóduje stream ze serveru a převádí jej na obraz. V případě, že divák zná i adresu přímo datového proudu sledované televize, může ji zadat do svého kompatibilního přehrávače a sledovat vysílání v něm, bez použití internetového prohlížeče.

Na stejném principu potom funguje i intranetová televize, která umožňuje, za použití počítače, členům lokální sítě sledovat streamované vysílání. To je velmi využívané například studenty na kolejích, kde některý počítač v lokální síti je vybaven většinou digitální (teoreticky lze i analogovou) televizní kartou, na kterou přijímá televizní signál a převádí jej na datové proudy. Ty jsou přenášeny po lokální síti a všichni studenti v této kolejší síti je mohou sledovat na svých počítačích.

2.4 Streaming

Streaming nebo také streamování je technologie, která zahrnuje získání datového proudu z nějakého audiovizuálního online (televizní karta) nebo offline (soubor uložený na disku) zdroje a kontinuální přenos proudu ke koncovému uživateli, kde je multimediální obsah ihned v reálném čase přehráván. Při sledování streamovaného obsahu nejsou za normálních okolností sledovaná data ukládána na disk a přijatý obsah tak není možno později znovu přehrát. Existují samozřejmě softwarové produkty, které ukládání přijatého streamovaného obsahu umožňují. Na druhé straně existují ale provozovatelé, kteří se snaží uživatelům zabránit v ukládání svého streamovaného obsahu, k čemuž využívají různé druhy šifrování nebo kódování multimediálního obsahu pomocí neveřejných kodeků. Takový obsah lze potom přehrát pouze v přehrávači získaném od provozovatele a pouze online.

Standardní streamování ovšem používá standardní audiovizuální kodeky a to hlavně z důvodu zmenšení potřebného objemu dat. Dříve zmíněný MPEG-2 nemá dostatečně vysoký kompresní poměr, a proto se častěji používá novější a výkonnější MPEG-4. Dalšími oblíbenými kodeky jsou především Windows Media, RealMedia, Real Time, Quicktime nebo Flash. Rozlišení streamovaného videa může být různé. Do nedávné doby se rozlišení pohybovalo nejčastěji kolem 320x240 bodů, což je pro sledování videa v celoobrazovkovém režimu velmi málo. Postupem času, díky zrychlování internetu, dochází ke zkvalitňování streamovaného videa, což vede ke zvyšování oblíbenosti internetové televize.

Streamování je možné s použitím několika protokolů, z nichž nejpoužívanější jsou UDP, TCP, RTP, RTSP a MMS.

Protokol UDP (User Datagram Protocol) přenáší proud formou řady paketů, což je jednoduché a efektivní. Nicméně tento protokol nemá žádný mechanismus, který by kontroloval, jestli datové pakety dorazily na místo určení. To znamená, že v případě chyby během přenosu není možné, aby si přijímací strana vyžádala znovuzaslání poškozených dat. Je tedy nutné využití technik samoopravných kódů, čímž se použití tohoto protokolu velmi podobá klasickému digitálnímu televiznímu vysílání. V případě výskytu neopravitelné chyby dojde k výpadku přenosu.

Protokol TCP (Transmission Control Protocol) garantuje korektní doručení každého bitu přenášeného streamu. To znamená, že v případě výskytu chyby si přijímací strana od strany odesílající vyžádá znovuzaslání poškozeného paketu. To ovšem může nějaký čas trvat a v případě okamžitého sledování videa v reálném čase by to znamenalo rušivé přerušování videa. To lze částečně vyřešit použitím vyrovnávací paměti, do které se přijímaný proud nejprve ukládá a až teprve po jejím naplnění se se zpožděním přehrává. V případě chyby je tedy více času na vyřešení problému. Použití vyrovnávací paměti samozřejmě způsobí prodlevu při spuštění a opoždění přehrávání oproti zdroji.

Protokol RTP (Real-time Transport Protocol) definuje standardizovaný paketový formát pro přenos audia a videa prostřednictvím internetu. Tento protokol je hojně využíván v internetové komunikaci, jako jsou videokonference a Voice over IP. Protokol RTP je většinou používán ve spojení s protokolem RTCP. Zatímco protokol RTP přenáší mediální proudy, protokol RTCP monitoruje přenosové statistiky a informace o kvalitě služby (QoS). Oba protokoly jsou založeny na protokolu UDP.

Protokol RTSP (Real Time Streaming Protocol) je protokol navržený pro použití v zábavních a komunikačních systémech k řízení streamovacích mediálních serverů. Jeho úkolem je navázat a řídit mediální relace mezi koncovými body. Příjemce přenosu může využívat příkazů, jako je play a pause, k ovládání přehrávání mediálních souborů na straně serveru. Ve většině případů se protokol RTSP používá hlavně k ovládání přehrávání. Vlastní přenos streamovaných dat bývá řešen protokolem RTP.

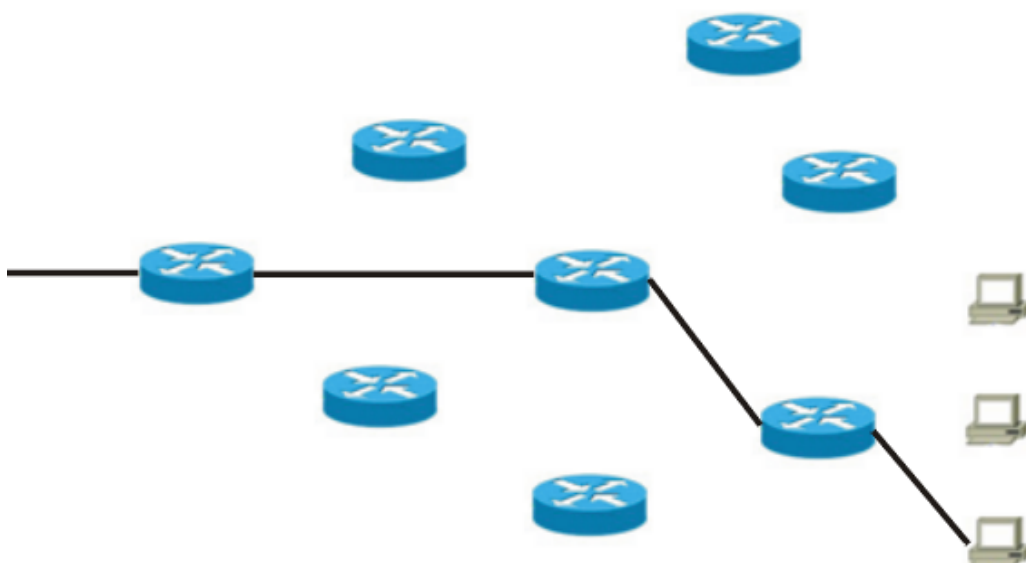
MMS (Microsoft Media Server) je název streamovacího protokolu společnosti Microsoft, který je používán pro přenos streamovaného obsahu metodou unicast. Může být realizován jak prostřednictvím protokolu UDP, tak i TCP.

Dle technologie zasílání paketů lze streamování ještě rozdělit na unicasting, broadcasting a multicasting.

2.4.1 Unicasting

Při této metodě je přenos zasílán pouze jedinému cíli. Tento typ přenosu z hlediska streamování není příliš vhodný. Používá se většinou jen u streamování Video on Demand (VoD), kde si každý uživatel vybere, který z mnoha pořadů chce sledovat a kdy je tudíž ke každému uživateli přenášén jiný obsah. V případě sledování online televize v reálném čase by ale při použití unicastu byl stejný mediální obsah přenášén k mnoha uživatelům formou mnoha vedle sebe přenášených stejných proudů, což by zbytečně vyžadovalo neúnosnou šířku pásma. Pro tyto účely je tedy mnohem vhodnější využití metody broadcast v případě místní sítě, nebo multicast v případě internetu.

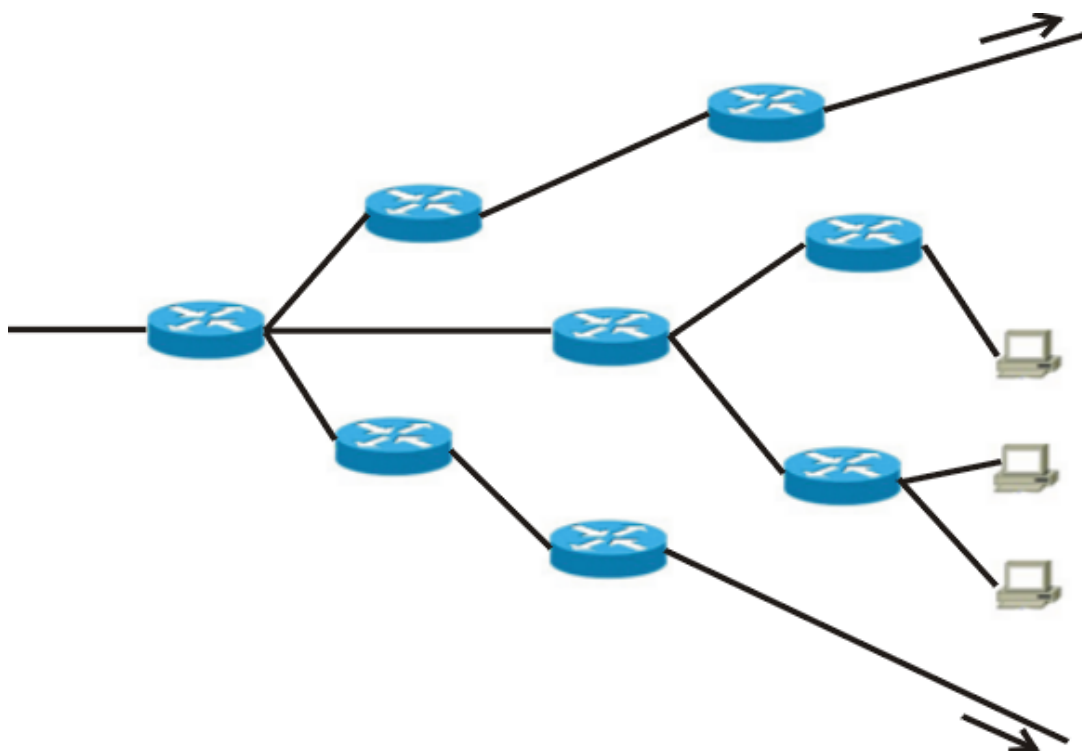
Obrázek č. 3: Unicasting



2.4.2 Broadcasting

Metoda broadcast zajišťuje přenos datových paketů ke všem zařízením dané sítě. Ne všechny počítačové sítě podporují tuto metodu. Nicméně u nás nejpoužívanější síť ethernet metodu broadcast podporuje a pro svou funkci ji i potřebuje. Nejvíce se broadcasting používá v lokálních sítích. Doplní-li se adresa lokální sítě, respektive její podsítě, samými logickými jedničkami, vznikne broadcastová adresa. Po zaslání datového paketu na broadcastovou adresu je paket zaslán na všechny adresy lokální sítě, respektive její podsítě. Z uvedeného vyplývá, že tato metoda způsobuje značný datový provoz, kdy při použití této metody při streamování je obsah datového proudu přenášen ke všem uživatelům dané lokální sítě, respektive podsítě, bez ohledu na to, jestli mají o tato data zájem. Proto je použití této metody omezeno jen na některé specifické situace.

Obrázek č. 4: Broadcasting



2.4.3 Multicasting

Metoda multicast je technologie určená pro doručení datového paketu určité skupině cílových zařízení zároveň. Z uvedených metod je nejmladší, což je její jediný handicap, protože některé starší síťové směrovače (routery) ji nepodporují. Metoda využívá velmi efektivně

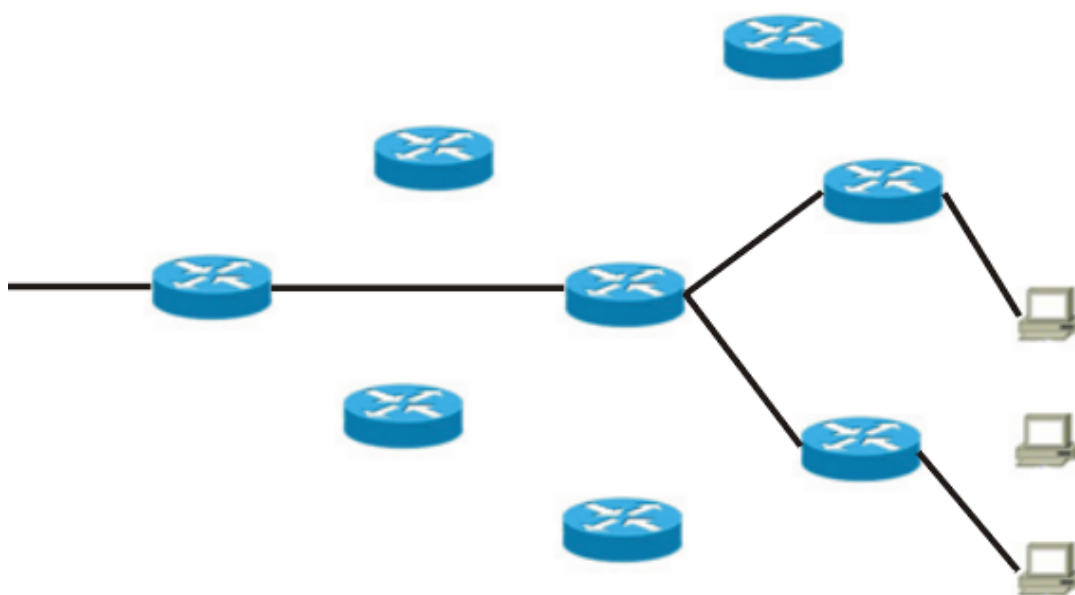
síťovou infrastrukturu, neboť datový paket je ze zdroje odeslán pouze jednou, přestože má být doručen velkému počtu cílových uživatelů. Metoda používá nejefektivnější strategii pro přenos dat, tak, aby každým uzlem sítě prošla data pouze jednou, přičemž pouze v bodě rozdělení cesty se v případě potřeby vytvoří kopie datového paketu. Nejběžnějším protokolem, který používá metodu multicast, je protokol UDP. Ten ovšem není zcela spolehlivý, protože nezaručuje doručení paketu do cíle, proto se vyvíjejí nové protokoly, jako je například Pragmatic General Multicast (PGM), které navíc obsahují funkci pro detekci ztráty paketu a možnost jeho opětovného zaslání.

Pro odesílání a příjem obsahu se používají multicastové skupinové IP adresy. Ty mají vyhrazený rozsah od 224.0.0.0 do 239.255.255.255. Odesílatel multicastu používá skupinovou adresu jako cílovou IP adresu odesílaných datových paketů. Příjemci multicastu použijí tutéž skupinovou adresu, aby informovali síť, že mají zájem přijímat datové pakety odesílané této skupině. Příjemci multicastu se tedy přihlásí k odběru paketů určených skupině a dokud se opětovně neodhlásí, budou jim pakety zasílány. Přihlašování a odhlašování od skupin se provádí pomocí protokolu Internet Group Management Protocol (IGMP).

Jakmile se k multicastovému vysílání přihlásí nějaký příjemce, je pro skupinu vytvořen multicastový distribuční strom. K tomu se nejčastěji používá Protocol Independent Multicast (PIM). Ten nastaví multicastový distribuční strom tak, aby datové pakety od odesílatele dorazily ke všem příjemcům, kteří se přihlásili do skupiny.

Při multicastovém vysílání odesílatel nepotřebuje znát adresy všech příjemců. Dokonce ani neví, kolik jich je. Data jsou doručena příjemcům na základě vytvořeného multicastového stromu konstruovaného síťovými uzly, které jsou blízko příjemcům, přičemž datový paket se kopíruje pouze v rozvětveních stromu, což zaručuje vysokou efektivitu sítě. To umožňuje příjem vysílání velkým počtem příjemců. Pro použití při streamování je tedy tato metoda nejvhodnější.

Obrázek č. 5: Multicasting



3 POŽADAVKY NA SYSTÉM

Hlavním účelem vytvářeného systému je, aby zaznamenával vysílání digitální televize. Uživatelé systému musí mít možnost zadat si předem čas a určený televizní kanál a systém poté určený program uloží na disk. Systém bude sloužit jako víceuživatelský programovatelný videorekordér.

3.1 Funkce systému

Systém bude spravován prostřednictvím webového rozhraní. Do něj se uživatel přihlásí a zadá si čas a televizní kanál, který chce nahrát. Systém bude evidovat nahrávací plány všech uživatelů a umožní jim jejich odstraňování či editaci. V nastavenou dobu systém obstará uložení daného streamu do souboru. Po uložení pořadu jej zobrazí v seznamu uložených pořadů každého uživatele a umožní jeho stažení, případně spuštění, a odstranění.

3.2 Prostředí systému

Bylo zadáno, že nahrávací systém má být implementován v prostředí OS Linux. Systém má být spravován pomocí webového rozhraní a vlastní ukládání záznamů na disk má být zprostředkováno programem v jazyku C/C++.

4 NÁVRH IMPLEMENTACE

Nahrávací systém se v souladu se zadáním dá rozdělit na dvě části: Část uživatelského systému, která bude sloužit ke správě systému pomocí webového rozhraní, a Část ukládání videa, která v daný čas obstará vlastní uložení daného televizního kanálu na disk. Tyto dvě části jsem se rozhodl zcela rozdělit. Nic nebrání tomu, aby Část uživatelského systému běžela na stejném počítači jako Část ukládání videa. To je ideální stav, ve kterém bude systém optimálně fungovat a poskytovat veškeré své funkce. Na druhé straně ale bude i možné, aby Část uživatelského systému běžela na zcela jiném serveru. To může být užitečné například v případě, kdy uživatel nevlastní veřejnou IP adresu a přesto chce, aby bylo možné do uživatelského systému přistupovat z internetu. V tom případě může umístit uživatelský systém na nějaký z internetu dostupný server – třeba i free webhosting. Uživatelé se budou moci do systému kdykoliv přihlásit a nastavovat si své nahrávací plány, nicméně nebudou si moci odtud stahovat nahrané pořady.

4.1 Část uživatelského systému

Je potřeba vytvořit webové rozhraní, prostřednictvím kterého by uživatelé mohli nastavovat své nahrávací plány. Rozhraní musí být multiuživatelské. Jeho implementace je možná pomocí různých programovacích jazyků. Je možné také využít podpory různých databází. Já jsem se rozhodl implementovat webové rozhraní pomocí jazyka PHP a namísto podpory databází využít ukládání dat do souborů XML. Pro tuto kombinaci jsem se rozhodl z důvodů zajištění minimálních technologických nároků.

V případě hostování webového rozhraní na počítači určeném zároveň pro ukládání videa, na kterém tedy běží operační systém Linux, stačí nainstalovat HTTP server, například Apache, a podporu PHP, přičemž webové rozhraní bude implementováno tak, aby stačila verze PHP 5. Některé distribuce OS Linux tyto součásti již nainstalované mají a u těch, které ne, je instalace otázkou několika málo minut, ne-li sekund.

V případě hostování webového rozhraní na nějakém veřejném serveru je stále mnohem jednodušší najít server s podporou PHP, než například ASP či JPS. Existuje i docela velké množství veřejných serverů s podporou PHP, které nabízejí hostování stránek zdarma. Většina těchto serverů sice nabízí i možnost využití databází – zpravidla Mysql, PostgreSQL nebo SQLite, nicméně rozhodl jsem se těchto možností nevyužít a v rámci minimalizace požadavků a maximálního zjednodušení instalace a nastavení webového rozhraní jsem místo databáze použil ukládání dat do souborů XML. To je do značné míry podporováno přímo jazykem PHP a pro potřeby tohoto uživatelského systému je tento přístup zcela dostačující.

4.1.1 Programovací jazyk PHP

PHP je široce využívaný univerzální skriptovací programovací jazyk, který byl původně navržen jako nástroj vývoje webu pro vytváření dynamických webových stránek. Pro tyto účely se PHP kód vkládá přímo do zdrojového HTML kódu dokumentu. HTML dokument se umístí

na server, který obsahuje modul PHP. V okamžiku, kdy chce uživatel dokument zobrazit ve svém prohlížeči, je kód PHP na straně serveru spuštěn a může provádět různé operace vedoucí většinou k vytvoření dynamického obsahu stránky, který se následně v prohlížeči uživateli zobrazí.

Jazyk PHP je použit na velkém množství různých stránek. Jeho prostřednictvím je naprogramován například projekt Wikipedie. Jeho oblíbenost tkví zejména v jeho jednoduchosti. Syntaxe jazyka vychází z několika programovacích jazyků (Perl, C, Pascal, Java) a díky kombinaci jejich vlastností nabízí vývojáři, který je s kterýmkoliv z nich seznámen, přátelské a známé prostředí.

4.1.1.1 Některé vlastnosti jazyka PHP

Jazyk PHP je dynamicky typový, což znamená, že datový typ proměnné se určí v okamžiku přiřazení hodnoty. Kvůli tomu má PHP dva operátory porovnání: při použití operátoru == dochází před porovnáním hodnot ke konverzi na stejný datový typ, při použití operátoru === je výraz pravdivý jen tehdy, jsou-li oba dva operandy stejného datového typu a jejich obsah má stejnou hodnotu.

Pole se dají indexovat číselnými indexy (jako v jazyce C), nebo mohou fungovat jako hashovací tabulka. Stejně pole může obsahovat oba typy indexů. Pole jsou heterogenní, což znamená, že v jednom poli mohou být prvky různých typů.

Řetězce lze uzavírat do uvozovek, v tom případě se při vyhodnocení provede nahrazení proměnných jejich hodnotami, nebo do apostrofů, v tom případě se nahrazují pouze escape sekvence „\“.

4.1.1.2 Výhody PHP

Za výhody jazyka PHP lze považovat například:

- existenci rozsáhlého souboru funkcí v základní knihovně PHP, včetně například používání technologie XML,
- nativní podporu mnoha databázových systémů,
- multiplatformost – PHP bez problémů funguje pod operačním systémem Linux i Windows,
- obrovskou podporu hostingových služeb – serverů nabízejících PHP hosting je opravdu mnoho,
- ohromující podporu komunity – lze nalézt spousty návodů či už hotových kódů, kterými se lze inspirovat nebo je i zdarma pro vlastní účely použít.

4.1.1.3 Nevýhody PHP

Jako nevýhodu jazyka PHP lze jmenovat například:

- jazyk PHP není nikde definován, je popsán pouze svou implementací,
- nekonzistentní notace názvů funkcí – v názvech některých víceslovných funkcí jsou jednotlivá slova oddělena podtržítkem, v jiných je použita tzv. velbloudí notace a v některých nejsou slova oddělena vůbec,
- ve standardní distribuci chybí ladicí nástroj (debugger),
- neexistuje žádné opravdu dobré vývojové prostředí srovnatelné například s MS Visual Studio.

4.1.2 Určení aktérů

Moje implementace uživatelského systému počítá se dvěma, respektive třemi druhy rolí:

- role Uživatel – může spravovat své nahrávací plány a uložené nahrávky,
- role Administrátor – má stejné možnosti jako Uživatel a k tomu ještě může spravovat uživatele,
- role Host nebo také neregistrovaný uživatel – do systému nemá přístup.

4.1.3 Seznam a popis procesů

1. Evidence uživatelů

zodpovědnost: Host

- 1.1 Žádost o registraci** – Systém žadatele vyzve, aby vyplnil některé osobní údaje a zvolil si heslo, načež je Administrátorovi zaslán mail, upozorňující na žádost o registraci. Pokud Administrátor registraci schválí, je uživateli zaslán informační mail, že jeho vstup do systému byl schválen.

zodpovědnost: Uživatel

- 1.2 Změna údajů** – Uživatel může editovat vlastní osobní údaje zadané při registraci včetně přihlašovacího jména a hesla.

zodpovědnost Administrátor

- 1.3 Schválení uživatele** – Administrátor může schválit žádost o registraci uživatele, čímž se z neregistrovaného uživatele stane právoplatný uživatel.
- 1.4 Zakázání uživatele** – Administrátor může určenému uživateli zakázat přístup do systému. O této skutečnosti je uživatel automaticky informován mailem.
- 1.5 Povolení uživatele** – Administrátor může povolit přístup do systému uživateli, jemuž byl dříve zakázán.
- 1.6 Založení uživatele** – Administrátor může založit nového uživatele.
- 1.7 Zrušení uživatele** – Administrátor může zcela zrušit vybraného uživatele.

2. Evidence položek plánu nahrávání

zodpovědnost Uživatel

- 2.1 Přidání položky do plánu** – Uživatel může zadat novou položku do plánu nahrávání.
- 2.2 Editace položky plánu** – Dříve zadané položky je možné upravovat – pokud nahrávání ještě neběží.
- 2.3 Zrušení položky plánu** – Položky nahrávacího plánu je možné zrušit. Pokud nahrávání dané položky již běží, její zrušení nebude mít na nahrávání vliv.

3. Evidence nahrávek

zodpovědnost Uživatel

- 3.1 Stažení nahrávky** – Uložené nahrávky je možné stáhnout na disk lokálního počítače.
- 3.2 Spuštění nahrávky** – Uložené nahrávky je možné přímo v prohlížeči přehrát. A to i v případě, že nahrávání dosud běží.
- 3.3 Zrušení nahrávky** – Své nahrávky uživatelé mohou po případném stažení či shlédnutí mazat.

4. Aktuality

zodpovědnost Uživatel

- 4.1 Zobrazení aktualit** – Po přihlášení jsou uživateli zobrazeny poslední aktuality (či různá upozornění).

zodpovědnost Administrátor

4.2 Přidání aktuality – Administrátor může přidávat aktuality či různá upozornění pro uživatele.

5. Log

zodpovědnost Administrátor

5.1 Zobrazení Logu – Administrátor si může zobrazit Log, do kterého systém ukládá všechny úspěšné i neúspěšné pokusy o přihlášení do systému včetně některých informací o uživateli – zejména jejich IP adresy.

4.1.4 Plánování nahrávek

Při plánování nahrávek mohou uživatelé využít různé možnosti, proto jsem se rozhodl tento proces popsat v této zvláštní kapitole.

Uživatelé si mohou nechat zobrazit svůj plán nahrávání. Na této stránce mohou přidávat, mazat i upravovat jednotlivé položky svého plánu. Pod seznamem položek plánu mají možnost rychlého přidání nové položky plánu. Zde musí zadat datum a čas začátku a konce nahrávání, televizní kanál, ze kterého se má nahrávat, název pořadu a frekvenci nahrávání. Frekvence může být „jednou“, „každý den“, „každý všední den“ a „každý týden“. V případě zvolení možnosti „jednou“ bude položka plánu po nahrání pořadu automaticky smazána. V případě zvolení ostatních frekvencí bude položka plánu po nahrání pořadu automaticky upravena pro nahrání dalšího pořadu. Když si uživatel zvolí například frekvenci „každý týden“, bude po nahrání pořadu datum jeho začátku a konce automaticky posunuto o týden.

Formulář pro rychlé zadání nového plánu je využit zároveň i pro editaci stávajících položek plánu. Pokud uživatel klikne na tlačítko „Edit“ u některé položky plánu, zobrazí se její údaje ve spodním formuláři, kde je uživatel může upravit dle potřeb a úpravy uložit.

Uživatelé ovšem mohou využít i mnohem pohodlnější způsob, jak přidat položky do svého plánu nahrávání. K tomu slouží v hlavní nabídce volba „Televizní program“. Po kliknutí na tuto volbu se uživateli zobrazí aktuální denní televizní program dostupných televizních kanálů. Uživatel si může zvolit den, jehož program si přeje zobrazit. U každého zobrazeného pořadu je zobrazen čas jeho začátku, název pořadu, pokud je k dispozici, tak podtitul, a v případě seriálů i číslo dílu, případně řady. Po najetí kurzoru myši na pořad se objeví okénko nápovědy s popisem pořadu – je-li k dispozici. Aktuálně běžící pořad je zvýrazněn. Podstatné je, že u každého pořadu jsou též dvě zaškrtnutá políčka. Jedno slouží k přidání pořadu do plánu nahrávání pro jednorázové nahrání pořadu. Pořad je tedy přidán do plánu jako další položka, je automaticky doplněn čas začátku a konce nahrávání, název pořadu i televizní kanál a frekvence nahrávání je nastavena na „jednou“. Druhé zaškrtnuté políčko naproti tomu slouží k přidání pořadu do plánu nahrávání pro opakované nahrávání pořadu. V tomto případě je do plánu nahrávání přidána položka stejným způsobem jako v předešlém případě s tím rozdílem, že frekvence je nastavena na „automaticky“. Takto přidané položky plánu nahrávání

se systém snaží automaticky aktualizovat. To probíhá tak, že jakmile začne nahrávání daného pořadu, systém se pokouší hledat v televizním programu stejného televizního kanálu další díl pořadu. V případě, že je u pořadu uvedeno číslo dílu, hledá se stejný pořad s následujícím číslem dílu (případně první díl další řady). To proto, aby se nenahrávaly stejné pořady vícekrát v případě repríz nebo aby se nemíchaly do sebe dvě různé řady nějakého seriálu v případě, že jsou na jednom televizním kanálu vysílány, což se běžně stává. V případě, že u pořadu číslo dílu uvedeno není, hledá se jednoduše první další pořad se stejným názvem. Po úspěšném nalezení dalšího dílu pořadu je pro něj vytvořena nová položka plánu nahrávání s nastavenou frekvencí nahrávání na „automaticky“, zatímco u právě nahrávané položky je frekvence změněna na „jednou“, čímž se zajistí, že se položka po nahrání pořadu smaže.

4.1.5 Struktura datových souborů

Veškerá data potřebná pro chod uživatelského systému jsou uložena v souborech XML. Pro chod systému jsou nezbytně potřebné dva soubory: soubor `users.xml`, kde jsou uloženy údaje o uživateli, a soubor `records.xml`, kde jsou uloženy údaje o plánech nahrávání. Další soubor `news.xml` je potřeba pro zobrazení aktualit a soubor `program.xml` obsahuje informace o televizním programu. Ten je systémem jednou denně aktualizován z projektu <http://xmltv.arcao.com>.

Obrázek č. 6: `Users.xml`

```
<users currentID="14">
  <user>
    <userID>1</userID>
    <username>Admin</username>
    <password>cb80cf23de362436596aa308cd3144a0</password>
    <fullname>Administrátor Pokusný</fullname>
    <role>admin</role>
    <email>nepiste_mi@seznam.cz</email>
    <lang>Czech</lang>
  </user>
</users>
```

Vysvětlení jednotlivých položek:

- atribut *currentID* funguje jako sequence v databázových systémech. Udává hodnotu ID pro příštího vytvořeného uživatele. Po vytvoření uživatele se inkrementuje,
- *userID* je identifikační číslo uživatele. Jak bylo dříve řečeno, uživatelé si mohou měnit svá přihlašovací jména. Aby to bylo možné, jsou jejich nahrávací plány vázány právě na ID uživatele,
- *username* je uživatelské a zároveň přihlašovací jméno,

- *password* je uživatelské heslo hešované pomocí funkce MD5,
- *fullname* je uživatelské celé jméno,
- *role* může být pouze admin pro Administrátora a user pro Uživatele,
- *email* obsahuje uživatelskou email, a
- *lang* je jazyk, ve kterém bude uživatelský systém s uživatelem komunikovat. Uživatelský systém je naprogramován vícejazyčně, všechny zobrazované textové řetězce jsou definovány jako konstanty v souborech `jazyk.php` v adresáři `lang\`. Prvek `lang` v XML souboru obsahuje jednoduše název souboru PHP, který se použije pro načtení konstant textových řetězců.

Obrázek č. 7: Schedule.xml

```
<records currentID="42">
  <record>
    <recordID>24</recordID>
    <userID>1</userID>
    <start>20100420183000</start>
    <stop>20100420185500</stop>
    <title>Simpsonovi</title>
    <channel>PRIMA COOL</channel>
    <episode>1.11</episode>
    <frequency>auto</frequency>
  </record>
</records>
```

Vysvětlení jednotlivých položek:

- Pro atribut *currentID* platí totéž co pro stejný atribut v případě souboru `users.xml`,
- *recordID* je ID dané položky plánu nahrávání,
- *userID* je ID uživatele, do jehož plánu nahrávání položka patří,
- *start* a *stop* jsou časy začátku a konce nahrávání,
- *title* je název pořadu,
- *channel* je název televizního kanálu,
- *episode* je číslo dílu, případně řady daného pořadu. V tomto případě se tedy jedná o první díl jedenácté řady, a
- *frequency* udává frekvenci nahrávání, ta je v tomto případě automatická.

4.1.6 Vzhled a vnitřní řešení uživatelského systému

Design stránek je pro dnešního uživatele velice důležitý. Uživatelé velice často dají na první dojem, který ze stránek mají a ten potom rozhodne o tom, jestli je stránky zaujmou či nikoliv. Já jsem se při tvorbě vzhledu rozhodl pro jednoduché konvenční řešení s logem systému umístěným do horní části stránky a menu umístěným vlevo. Design je vytvořen prostřednictvím šablony stylů `style.css`, takže v budoucnu není problém jej jakkoliv upravit.

Vlastní uživatelský systém sestává z několika souborů `php`. Každá stránka systému má vlastní `php` skript, na který odkazují položky v menu či jiné odkazy. V každém `php` skriptu je vlastní funkčnost stránky, přičemž veškerý výstup je ukládán do proměnné `$content`. Nakonec je vložen skript `page.php`, který zprostředkuje vlastní vykreslení stránky. Ten ještě vkládá skript `menu.php`, který do proměnné `$menu` vloží svůj výstup pro menu. Menu bude rozdílné podle toho, zda je přihlášen nějaký uživatel a jestli je to Uživatel nebo Administrátor. Skript `page.php` poté vykreslí na stránku logo, menu z proměnné `$menu` i obsah stránky z proměnné `$content`. Další komunikace mezi skripty probíhá pouze s použitím metody `POST`.

Ještě bych na tomto místě měl zmínit skript `get_records.php`, který neprovádí žádnou pro uživatele viditelnou funkcionalitu, ale je naprosto klíčový pro funkčnost celého nahrávacího systému. Jeho výstupem je totiž text obsahující informace o tom, co se má právě nahrávat, a těmito informacemi se bude řídit druhá část projektu, zajišťující vlastní nahrávání videa.

4.2 Část ukládání videa

Je potřeba vytvořit program v jazyce `C/C++`, který bude ukládat na disk pořady vybrané uživateli. Vzhledem k tomu, že jsem projekt rozdělil na dvě části a Část uživatelského systému převzala většinu funkcionality celého systému, bude tento program docela jednoduchý. Jeho úkolem bude jen načíst informace o tom, jestli se má právě něco nahrávat, v případě, že ano, spustit nahrávání, kontrolovat, jestli běží, dokud má a pak jej ukončit. Program poběží v nekonečné smyčce a naprostou většinu doby svého běhu bude neaktivní.

4.2.1 Rozbor programu

Program si bude v paměti evidovat záznamy, které se mají nahrávat. Většinu evidovaných údajů získá od skriptu `get_records.php` z uživatelského systému. Odtud získá ID plánu nahrávání, ID uživatele, který chce záznam nahrávat, čas začátku a čas konce nahrávání, televizní kanál, ze kterého se má záznam nahrávat, název pořadu a případně číslo dílu. Mimo těchto údajů se bude pro každý záznam evidovat ještě `PID`, což je identifikátor procesu, pod kterým bude nahrávání probíhat.

4.2.2 Průběh programu

- 1) Stáhne se výsledek skriptu `get_records.php`. Pokud je uživatelský systém uložen přímo na tomtéž PC, pak se stáhne z místního hostitele (localhost), pokud je na nějakém jiném serveru, stáhne se z něj.
- 2) Pokud skript vrátí nějakou položku k nahrávání, je podle *recordID* zjištěno, jestli toto nahrávání program již eviduje. Pakliže ne, je záznam přidán do seznamu záznamů a je mu přidělen $PID = -1$.
- 3) Proběhne kontrola nahrávání záznamů. Projde se seznam všech nahrávání a u každého záznamu seznamu se zkontroluje jeho PID:
 - pokud $PID = 0$, nahrávání záznamu bylo již ukončeno. Takový záznam se odstraní ze seznamu.
 - pokud $PID > 0$, nahrávání záznamu by mělo probíhat. V tom případě se zkontroluje, jestli proces s daným PID běží. Dále se podle evidovaného konce nahrávání zjistí, jestli by se nahrávání nemělo zastavit. Pokud ano, tak se pomocí funkce *kill* ukončí proces s daným PID a PID se změní na hodnotu 0.
 - pokud $PID = -1$, záznam se ještě nenahrává. V tom případě, pokud do začátku nahrávání zbývá méně než dvě minuty, je zavolána metoda pro spuštění nahrávání. Ta vytvoří nový proces pomocí funkce *fork*. Rodičovský proces uloží PID procesu potomka do evidence záznamů. Nově vytvořený proces čeká do chvíle, kdy se má spustit nahrávání, načež se pomocí funkce *exec* spustí program *wget*, který začne ukládat patřičný stream do zadaného souboru.
- 4) Program čeká 30 sekund, načež se vrací do bodu 1).

4.3 Spolupráce Části uživatelského systému s Částí ukládání videa

Jak již bylo řečeno, mohou být obě části umístěny na stejném počítači nebo může být Část uživatelského systému umístěna na jiném veřejně dostupném serveru. Ve skutečnosti server nemusí být dostupný pro veřejnost, ale musí být dostupný pro své uživatele a zároveň pro počítač ukládající videa, který z něj musí mít možnost získávat potřebné údaje o plánech nahrávání. Pokud počítač ukládající videa není dostupný pro Část uživatelského systému (typicky v případě, kdy počítač ukládající videa nemá veřejnou IP adresu), uživatelé si nebudou moci na stránkách projektu spouštět a stahovat uložené záznamy. Stahování uložených záznamů musí být v tom případě řešeno jinak – například jejich uploadováním na nějaký pro všechny dostupný FTP server. Toto je ovšem jediné omezení. Jinak mezi použitím obou částí dohromady nebo odděleně není žádný rozdíl.

Program uskutečňující ukládání záznamů každých zhruba 30 sekund spustí skript `get_records.php`, který mu jako výsledek vrátí seznam záznamů, které by se měly nahrávat. Do tohoto seznamu jsou záznamy vkládány s předstihem, takže jsou tam i záznamy, které se mají začít nahrávat v blízké budoucnosti. Zároveň skript upravuje čas konce nahrávání tak, že přidá určitý počet minut z důvodu skluzu televizního programu.

Skript `get_records.php` zároveň obstará aktualizaci plánů nahrávání, což zahrnuje odstranění již nahraných položek plánu i případné hledání dalších dílů položek s automatickou frekvencí. Hledání dalších dílů se musí provádět hned po začátku nahrávání, protože se může stát, že jsou v televizním programu dva díly pořadu hned za sebou, což vzhledem k tomu, že konec každého nahrávání je posunut kvůli skluzu televizního programu, znamená, že následující díl pořadu se musí začít nahrávat v době, kdy se předchozí díl ještě nahrává.

Program uskutečňující ukládání záznamů zpracuje výsledek skriptu `get_records.php`, nové položky k uložení přidá do svého seznamu a obstará jejich uložení.

Výstup skriptu `get_records.php` vypadá jako konfigurační soubor používaný v systému Linux. Na konec každého řádku je přidán komentář obsahující tag `
`, což program ignoruje, nicméně je to vhodné pro zobrazení skriptu v internetovém prohlížeči, který díky tomu nenapiše vše do jednoho řádku.

Obrázek č. 8: Příklad výstupu skriptu `get_records.php` zobrazený v prohlížeči

```
NEW_RECORD_ITEM #
recordID = 198 #
userID = 1 #
start = 1271875500 #
stop = 1271877300 #
title = Dva a půl chlapa #
channel = NOVA #
episode = 16.6 #
#
NEW_RECORD_ITEM #
recordID = 199 #
userID = 1 #
start = 1271877000 #
stop = 1271880600 #
title = Dr. House #
channel = NOVA #
episode = 23.5 #
#
```

5 INSTALACE PROJEKTU

Instalace a konfigurace projektu se opět dělí na dvě části: na instalaci a konfiguraci Části uživatelského systému a na instalaci a konfiguraci Části ukládání videa.

5.1 Instalace a konfigurace Části uživatelského systému

V případě instalace Části uživatelského systému na stejný počítač, na kterém bude i Část ukládání videa, je nutné, aby v počítači byl nainstalován operační systém Linux a HTTP server s podporou PHP. Já jsem celý projekt vytvořil, odladil a testoval na operačním systému Linux Debian 5.0 v minimální instalaci s doinstalovaným HTTP serverem Apache2 ve verzi 2.2.9-10 a balíčkem PHP5 ve verzi 5.2.6. V případě instalace na hostingový server je nutné pouze to, aby server podporoval PHP5.

Instalace probíhá velmi jednoduše. Stačí jen do kořenové složky webu nakopírovat všechny soubory php, soubor style.css, obstarávající vzhled webu, adresář images, obsahující obrázky potřebné pro chod webu, adresář data, obsahující datové xml soubory a adresář lang, obsahující jazykové mutace webu. Soubory v adresáři data by měly mít jako jediné nastavení atribut povolující zápis do souborů.

Konfigurace webové části probíhá prostřednictvím editace souboru config.php. V něm je možné nastavit umístění souborů XML, zadat url adresu, ze které se bude stahovat televizní program a nastavit názvy všech televizních kanálů, jak se budou zobrazovat a jejich kódů, které jsou použity v televizním programu. Mimo to jsou tam ještě dvě důležité proměnné:

- proměnná *\$additionMinutes* udává o kolik minut bude každý pořad prodloužen kvůli skluzu televizního programu. Tato hodnota je přidána k času ukončení nahrávání každého záznamu skriptem `get_records.php`. Ve výchozím nastavení je tato hodnota nastavena na 5 minut, což se během testování ukázalo, jako dostatečné. Samozřejmě, že různé televizní kanály se v přesnosti svých televizních programů mohou značně lišit, takže v případě potřeby je možné hodnotu adekvátně přizpůsobit,
- proměnná *\$additionLeading* udává, s jak velkým předstihem (v minutách) před vlastním začátkem nahrávání bude skript `get_records.php` daný plán nahrávání vypisovat. V případě, že je celý nahrávací systém nainstalován na jednom počítači, může být tato hodnota docela malá, i když by neměla být menší než jedna minuta. V případě, že je Část uživatelského systému nainstalována na jiném serveru, může zvýšení této hodnoty kompenzovat jeho nespolehlivost. Pokud by byl server extrémně nespolehlivý nebo třeba i při problémech s připojením počítače zajišťujícího nahrávání, je možné navýšit tuto hodnotu třeba i na několik hodin. V takovém případě by se do evidence plánů nahrávek nahrávacího programu dostaly všechny nahrávky s několikahodinovým předstihem a případný výpadek spojení se serverem obsahujícím uživatelský systém by na stávající plány nahrávek neměl negativní vliv. Ve výchozím nastavení je tato hodnota nastavena na 5 minut, což je

hodnota vhodná pro případ obou částí systému na stejném počítači a zároveň dostatečná i pro použití uživatelského rozhraní na jiném serveru, za předpokladu, že je daný server standardně spolehlivý.

5.2 Instalace a konfigurace Části ukládání videa

Instalace této části je velmi jednoduchá. Stačí do určené složky systému Linux nakopírovat soubor zdrojového kódu programu `recorder.cpp`, konfigurační soubor `recorder.conf` a soubor `Makefile`. Potom v této složce zavolat příkaz *make*, který obstará kompilaci programu. Vytvoří se spustitelný soubor `recorder`. Ten po svém spuštění již bude ukládat naplánované pořady. Všechny důležité události zapisuje do souboru `recorder.log`. Pro správnou funkci programu bude patrně nutné upravit konfiguraci v souboru `recorder.conf`.

Soubor `recorder.conf` obsahuje několik důležitých nastavení. Nejdůležitější je patrně nastavení klíče *records_source*, který musí obsahovat kompletní url adresu souboru `get_records.php`. Dalším velmi důležitým klíčem je *records_dir*, který musí obsahovat cestu udávající, kam se mají ukládat nahrávané video soubory. Poslední dva klíče jsou *channels_names* a *channels_ips*. Klíč *channels_names* musí obsahovat názvy televizních kanálů, oddělené čárkou. Tyto názvy se musí shodovat s názvy televizních kanálů tak, jak jsou zadány v konfiguračním souboru Části uživatelského systému. A poslední klíč *channels_ips* obsahuje adresy streamů televizních kanálů, oddělené čárkou, přičemž pořadí adres musí korespondovat s pořadím názvů kanálů.

Obrázek č. 9: Recorder.conf

```
#
# konfigurační soubor programu recorder
#
# cesta ke skriptu:
records_source =http://192.168.1.137/get_records.php

# cesta pro ukládání:
records_dir = /var/www/records

# názvy televizních kanálů (s mezerami si není třeba dělat starosti)
channels_names = NOVA CINEMA , PRIMA, PRIMA COOL , BARRANDOV

# adresy korespondujících streamů
channels_ips = 192.168.1.137:9004,192.168.1.137:9005,192.168.1.137:9006,192.168.1.137:9007
```

6 ZÁVĚR

Cílem mé bakalářské práce bylo vytvoření systému pro záznam digitální televize. Systém měl umožnit uživatelům správu svých nahrávacích plánů i uložených nahrávek. To můj systém splňuje. Při jeho tvorbě jsem kladl hlavní důraz na univerzálnost, maximální použitelnost a s tím související minimální požadavky systému na hardware i software. Celý projekt jsem rozdělil na dvě maximálně na sobě nezávislé části, čímž jsem systém zpřístupnil uživatelům, kteří nevlastní veřejnou IP adresu.

Část webového uživatelského rozhraní může fungovat na jakémkoliv hostingu, který podporuje technologii PHP5. Žádná jiná technologie (databáze, AJAX, JavaScript) není pro chod rozhraní potřeba. Uživatelé si po přihlášení mohou spravovat své plány nahrávání, mohou si z televizního programu pohodlně vybrat pořady, které chtějí jednorázově nahrát, i seriály, u nichž chtějí nahrávat všechny díly. V takovém případě dokáže systém nahrát všechny díly seriálu i tehdy, když se nevysílají pravidelně. Nahrané pořady je možné z rozhraní stáhnout do lokálního počítače nebo spustit přímo v prohlížeči bez nutnosti stahování. Spustit v prohlížeči se dají i nahrávky, které se právě nahrávají, systém tedy může fungovat i jako tzv. Time Shifting.

Část ukládající záznamy je naprogramována pro prostředí OS Linux v jazyku C++. Pro její správnou funkci postačí pouze minimální distribuce OS Linux s nainstalovaným programem wget. Samozřejmě je také potřeba mít k dispozici streamy programů televizních stanic, ze kterých se bude nahrávat. Je jedno, jestli ke streamování dojde ve stejném PC, například z digitální televizní karty pomocí programu Getstream, nebo jestli se bude ukládat stream vysílaný odněkud z místní sítě či z internetu.

Instalace i konfigurace celého mého systému je velice jednoduchá. Stačí jen nahrát soubory na server, jediným příkazem zkompileovat program pro ukládání záznamů, jednoduše zadat názvy televizních stanic a adresy příslušných streamů a potom již stačí jen zaškrtnout oblíbené pořady.

SEZNAM POUŽITÉ LITERATURY

Monografie:

- [1] BEDNÁŘ, Jiří; GREGORA, Pavel. *Přijem DVB-T*. 1. vyd. Praha : BEN – technická literatura, 2007. 134 s. ISBN 978-80-7300-221-3.
- [2] KOSEK, Jiří. *PHP a XML*. 1. vyd. Praha : Grada, 2009. 367 s. ISBN 978-80-247-1116-4.
- [3] LEGÍŇ, Martin. *Televizní technika DVB-T*. 1. vyd. Praha : BEN – technická literatura, 2006. 286 s. ISBN 80-7300-204-3.

Internetové zdroje:

- [4] *DigiPrijem.cz* [online]. 2010 [cit. 2010-04-26]. Dostupné z WWW: <<http://www.digiprijem.cz>>.
- [5] *Iptv.digizone.cz* [online]. 2010 [cit. 2010-04-26]. Co je IPTV a v čem se liší od kabelové televize. Dostupné z WWW: <<http://iptv.digizone.cz/co-je-iptv-a-v-cem-se-lisi-od-kabelove-televize>>.
- [6] MPEG transport stream#PMT In *Wikipedia: the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 11. 2. 2005, 20. 4. 2010 [cit. 2010-04-26]. Dostupné z WWW: <http://en.wikipedia.org/wiki/MPEG_transport_stream#PMT>.
- [7] Multicast In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 25.2.2002, 25.3.2010 [cit. 2010-04-26]. Dostupné z WWW: <<http://en.wikipedia.org/wiki/Multicast>>.
- [8] *Networking Tutorial* [online]. 2010 [cit. 2010-04-26]. Network Broadcasting and Multicasting. Dostupné z WWW: <<http://www.comptechdoc.org/independent/networking/guide/netbroadcasting.html>>.
- [9] PHP In *Wikipedia : the free encyclopedia* [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2. 6. 2004, 12. 4. 2010 [cit. 2010-04-26]. Dostupné z WWW: <<http://cs.wikipedia.org/wiki/PHP>>.

SEZNAM OBRÁZKŮ

Obrázek č. 1: Paket transportního proudu	6
Obrázek č. 2: Hlavička paketu transportního proudu.....	6
Obrázek č. 3: Unicasting	10
Obrázek č. 4: Broadcasting.....	11
Obrázek č. 5: Multicasting.....	12
Obrázek č. 6: Users.xml	19
Obrázek č. 7: Schedule.xml	20
Obrázek č. 8: Příklad výstupu skriptu get_records.php zobrazený v prohlížeči.....	23
Obrázek č. 9: Recorder.conf	25